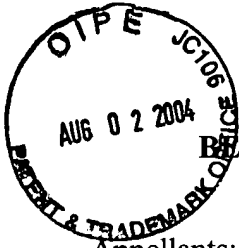


AF 2154/120

AUG 0 2 2004 TRANSMITTAL OF APPEAL BRIEF (Large Entity)				Docket No. POU920000019US1	
In Re Application Of: Uceda-Sosa et al.					
Application No. 09/584,609	Filing Date 05/31/2000	Examiner Mohammad A. Siddiqi	Customer No. 23405	Group Art Unit 2154	Confirmation No. 2859
Invention: METHOD, SYSTEM AND PROGRAM PRODUCTS FOR CONCURRENTLY ACCESSING A GLOBAL DATA REPOSITORY BY MULTITHREADED CLIENTS					
<u>COMMISSIONER FOR PATENTS:</u>					
Transmitted herewith in triplicate is the Appeal Brief in this application, with respect to the Notice of Appeal filed on June 1, 2004					
The fee for filing this Appeal Brief is: \$330.00			RECEIVED AUG 0 5 2004 Technology Center 2100		
<input type="checkbox"/> A check in the amount of the fee is enclosed.					
<input type="checkbox"/> The Director has already been authorized to charge fees in this application to a Deposit Account.					
<input checked="" type="checkbox"/> The Director is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. 09-0463 (IBM)					
<u>Blanche E. Schiller</u> Signature			Dated: July 30, 2004		
Blanche E. Schiller, Esq. Reg. No. 35,670 Heslin Rothenberg Farley & Mesiti P.C. 5 Columbia Circle Albany, NY 12203 Telephone: (518) 452-5600 Facsimile: (518) 452-5579			<div>I certify that this document and fee is being deposited on July 30 2004 with the U.S. Postal Service as first class mail under 37 C.F.R. 1.8 and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.</div> <div><u>Jill K. Becker</u> Signature of Person Mailing Correspondence</div> <div>Jill K. Becker Typed or Printed Name of Person Mailing Correspondence</div>		
CC:					



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Appellants: Uceda-Sosa et al.

: Group Art Unit: 2154

Serial No.: 09/584,609

: Examiner: Mohammad A. Siddiqi

Filed: 05/31/2000

: Appeal No.:

Title: METHOD, SYSTEM AND PROGRAM PRODUCTS FOR CONCURRENTLY
ACCESSING A GLOBAL DATA REPOSITORY BY MULTITHREADED CLIENTS

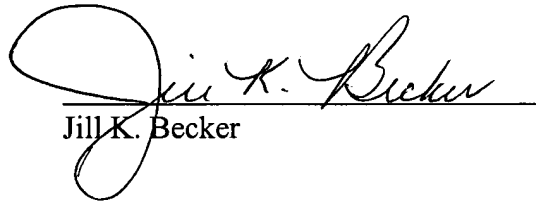
RECEIVED

Certificate of Mailing

AUG 05 2004

Technology Center 2100

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as first class mail in an envelope addressed to: Mail Stop Appeal Briefs – Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on July 30, 2004.



Jill K. Becker

Date of Signature: July 30, 2004.

Mail Stop Appeal Briefs – Patents
Commissioner for Patents
P.O. Box 1450, Alexandria, VA 22313-1450

Brief of Appellants

Dear Sir:

This is an appeal from a final rejection, dated January 30, 2004, rejecting claims 1, 3, 5, and 7-24, all the claims being considered in the above-identified application. The Brief is due within two months from the date the Notice of Appeal was received at the United States Patent and Trademark Office. Since Appellants' postcard indicates that the Notice of Appeal was received on June 1, 2004, this Brief is being timely filed before August 1, 2004. This Brief is

08/03/2004 EAREGAY1 00000006 090463 09584609

01 FC:1402 330.00 DA
POU920000019US1

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

accompanied by a transmittal letter authorizing the charging of appellants' deposit account for payment of the requisite fee set forth in 37 C.F.R. 1.17(c).

Real Party in Interest

This application is assigned to International Business Machines Corporation by virtue of an assignment executed by the co-inventors on September 6, 2000, September 7, 2000 and September 26, 2000, and recorded with the United States Patent and Trademark Office at reel 011236, frame 0137, on October 23, 2000. Therefore, the real party in interest is International Business Machines Corporation.

Related Appeals and Interferences

To the knowledge of the appellants, appellants' undersigned legal representative, and the assignee, there are no other appeals or interferences which will directly affect or be directly affected by or have a bearing on the Board's decision in the instant appeal.

Status of Claims

This patent application was filed on May 31, 2000 with the United States Patent and Trademark Office. As filed, the application included three (3) independent claims (i.e., claims 1, 3 & 5) and three (3) dependent claims (i.e., claims 2, 4 & 6). On April 12, 2002, appellants filed a Preliminary Amendment that amended independent claims 1, 3 and 5 and added dependent claims 7-24.

In an initial Office Action, dated August 6, 2003, claims 1-24 were rejected under 35 U.S.C. 102(e) as being unpatentable over Traversat et al. (U.S. Patent No. 6,119,129; hereinafter, "Traversat"). In appellants' response, dated December 8, 2003 (with which a request for a one-

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

month extension of time and requisite fee were enclosed), claims 1, 3, 5, 7, 10-13, 16-19 & 22-24 were amended, and claims 2, 4 & 6 were canceled.

In a second and final Office Action, dated January 30, 2004, claims 1, 3, 5 & 7-24 were rejected under 35 U.S.C. 103(a) as being unpatentable over Traversat in view of Jiang et al. (U.S. Patent 6,453,354; hereinafter, "Jiang"). Appellants timely filed a response to the second and final Office Action on April 26, 2004, in which no claims were amended. Appellants received an Advisory Action, dated May 13, 2004, which indicated that appellant's Response to the final Office Action did not place the application in condition for allowance.

A Notice of Appeal to the Board of Patent Appeals and Interferences was filed on May 28, 2004, accompanied by a Petition for Extension of Time to File Notice of Appeal, in which a one-month extension of time was requested, and the requisite fee. The Notice of Appeal and Petition for Extension of Time were received at the United States Patent and Trademark Office on June 1, 2004. The status of the claims is therefore as follows:

Claims allowed – none;

Claims objected to – none;

Claims rejected – 1, 3, 5 & 7-24; and

Claims canceled – 2, 4 & 6.

Appellants are appealing the rejection of claims 1, 3, 5 & 7-24.

Status of Amendments

No claim amendment was effectuated by the Response to Final Office Action, dated April 26, 2004. The claims as set out in Appendix A include all prior entered amendments.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

Summary of Invention

The present invention is directed, in one aspect, to managing the accessing of one or more global data repositories of a distributed computing environment by multithreaded client applications (page 5, lines 12-16; and page 3, lines 9-12).

More particularly, in one aspect, the present invention is a method (claim 1), system (claim 3), and program storage device, embodying a program of instructions readable and executable by a machine to perform the method (claim 5), for managing the locking of resources of a global data repository of a distributed computing environment (page 5, lines 12-18). The method includes, for instance, issuing a request, via a thread 600 of a multithreaded client application 602 of the distributed computing environment, for a lock on a resource (e.g., Tables A, B) of the global data repository (FIG. 6; page 15, lines 4-14); and obtaining the requested lock 606 for the thread independent of the threading model of the operating system of the distributed computing environment (page 15, lines 6-10; and page 14, line 27 to page 15 line 3). A local tree 604 is employed in obtaining the lock for the thread 600 (page 15, lines 4-14; and FIG. 6). This local tree 604 is local to the client application 602 (FIG. 6; FIG. 5; and page 11, lines 10-14) and has a mount point 608 that is usable by the client application 602 to lock the resource (FIG. 6; and page 15, lines 4-20). The resource is also lockable via another mount point 613 of the local tree 604 or another local tree 614 (FIG. 6; and page 15, line 21 to page 16, line 2).

In another aspect of the invention (see claims 12, 18 and 24), the method for managing the locking of resources of a global data repository additionally includes use of the resource, for which a lock has already been obtained for a thread 600, by another thread 612 of the multithreaded client application 602 (FIG. 6; page 15, lines 4-20; and page 13, lines 11-18).

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

Issues

1. Whether claims 1, 3, 5 & 7-24 were rendered obvious under 35 U.S.C. 103(a) by Traversat in view of Jiang.

Grouping of Claims

There is one ground of rejection, and thus, one group of claims, Group I. Group I includes claims 1, 3, 5 & 7-24; however, the claims of Group I do not stand or fall together. Instead, each of the following subgroups of Group I includes claims that provide a separate basis of patentability.

- (i) Claims 1, 3, 5, 7-11, 13-17 & 19-23; and
- (ii) Claims 12, 18 & 24.

As understood, the claims of one subgroup of claims do not stand or fall with any other subgroup of claims, except that, if the claims of Subgroup (i) stand, then the claims of Subgroup (ii) also stand because they depend from the independent claims of Subgroup (i). However, appellants respectfully submit that none of the subgroups fall together because the claims of each subgroup provide a separate basis of patentability.

Argument

Claims 1, 3, 5 & 7-24, i.e. the claims of Subgroups (i) and (ii) of Group I, stand rejected under 35 U.S.C. 103(a) as being unpatentable over Traversat in view of Jiang.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

Group I, Subgroup (i): Claims 1, 3, 5, 7-11, 13-17 & 19-23

As noted, the claims of Group I, Subgroup (i), stand rejected under 35 U.S.C. 103(a) as being unpatentable over Traversat in view of Jiang. Appellants respectfully submit that the rejection of this subgroup is erroneous for the reasons below, and therefore, respectfully request reversal of this rejection.

In one aspect, appellants' invention is directed to multithreaded clients accessing resources of a global data repository. For example, appellants claim a method of managing the locking of resources of a global data repository of a distributed computing environment (e.g., claim 1). The method includes, for instance, issuing a request, via a thread of a multithreaded client application of the distributed computing environment, for a lock of a resource of the global data repository; and obtaining the lock for the thread independent of a threading model of an operating system of the distributed computing environment. The obtaining includes employing a local tree in obtaining the lock, the local tree being local to the client application and having a mount point usable by the client application to lock the resource. The resource is further lockable via another mount point of the local tree or another local tree.

Advantageously, employing local trees allows applications to access a resource (e.g., a table) via a mount point (e.g., directory) of a local tree even though the resource is locked in a global data repository (e.g., global database). Further, the same resource (table) can be mounted and locked through different mount points (directories). For instance, with a global database that includes Table X and a local tree that includes directories A and B, an application can mount and lock Table X through Directory A and mount and lock Table X through Directory B. Therefore, in this example, Table X is mounted and locked more than once, as if it were two different resources.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

The use of a local tree is recited in appellants' claimed invention. For example, appellants' claim 1 includes a local tree that is local to a client application and has a mount point usable by the client application to lock a resource, the resource also being lockable via another mount point of either that local tree or another local tree. The use of a local tree that is local to a client application is not taught or suggested by Traversat or Jiang, either alone or in combination.

For example, Traversat describes a journaling mechanism that maintains a detailed journal of all modifications made to a global configuration database (see Abstract thereof). In the context of this journaling mechanism, Traversat discloses locks relative to a global tree structure in a global configuration database (see, e.g., col. 8, lines 21-24 and lines 63-67). A lock in Traversat is globally applied to a resource so that database transactions are provided with one view of what is globally accessible in the database (see, e.g., col. 8, lines 50-67). Traversat does not teach or suggest using a local tree that is local to a client application. Instead, in Traversat, the tree structure being locked in the configuration database is a global tree, rather than a local tree, and resides on a server which is separate from client machines and client applications (see col. 8, lines 28-29 and FIG. 1 thereof). Traversat explicitly avoids the use of trees local to client applications. Thus, the tree in Traversat is not local to a client application, as recited by the claims presented herewith.

Traversat also fails to teach or suggest that the local tree, which is local to the client application, has a mount point usable by the client application to lock a resource, the resource also being lockable via another mount point of either that local tree or another local tree.

The failure of Traversat to teach or suggest one or more aspects of appellants' claimed invention is admitted in the Final Office Action. The Final Office Action explicitly states on page 3:

Traversat is silent about "Wherein said obtaining comprises employing a local tree in obtaining said lock, said local tree being local to the client application and having a mount point usable by

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

the client application to lock said resource, and wherein said resource is further lockable via another mount point of one of said local tree and another local tree.”

Since Traversat fails to teach or suggest one or more aspects of appellants’ claimed invention, Traversat is combined with Jiang. However, Jiang does not overcome the deficiencies of Traversat. That is, appellants respectfully submit that Jiang also fails to describe, teach or suggest one or more aspects of appellants’ claimed invention.

In general, Jiang describes a method of operating a data mover computer using a connection-oriented protocol for permitting a client to access a file system (col. 3, lines 59-62). In particular, the cited portion of Jiang describes a two-level locking scheme for obtaining file locks (col. 27, lines 23-34) in a networked computing environment wherein data mover computers are employed to service client access requests to file systems (col. 3, line 59 to col. 4, line 2).

There is a significant difference between the claimed invention and the acquisition of file locks described in Jiang. For instance, as recited in appellants’ claim 1, the obtaining of a lock for a resource comprises employing a local tree in obtaining said lock, said local tree being local to the client application. Appellants respectfully submit that Jiang does not teach or suggest, at the very least, employing a local tree that is local to a client application. Although a local tree is mentioned in Jiang (Col. 3, line 40-45), the local tree in Jiang is local to a server and not to the client application, as claimed by appellants. This is explicitly stated in Jiang: “The server attempts to resolve the requested name to a file contained within the local directory tree....” Thus, the tree in Jiang is local to the server and not to the client application.

In Jiang, client requests are sent to the server and the server services those requests (see, e.g., Col. 3, lines 30-55). The requests are serviced using trees local to the server and not trees local to the client application. The server in Jiang operates like a file system of an operating

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

system to resolve names. Jiang fails to teach or suggest employing a tree local to a client application, the same claimed element missing from Traversat. Since Jiang and Traversat both fail to teach or suggest employing a local tree in obtaining a lock, the local tree being local to the client application, the combination of Jiang and Traversat also fails to teach or suggest this claimed element.

In addition, appellants respectfully submit that the combination of Traversat and Jiang is improper, and thus, respectfully request withdrawal of the rejection for this reason.

Traversat explicitly avoids the use of trees local to client applications. Traversat specifically states at Col. 4, lines 10-16:

In the described embodiment, data relating to client machine and user configuration in a network is stored in on [sic] a server as part of a server JSD. The configuration information for each client, also referred to as subsystem, is stored in the server schema. This is in contrast to conventional networks where configuration information regarding a client is hardcoded or stored on the client machine.

Because Traversat teaches away from using trees local to a client application, there would have been no motivation to combine Traversat with a reference that presumably teaches the use of local trees. Although appellants respectfully, but most strenuously, submit that Jiang does not teach or suggest the use of trees local to a client application, if it was believed that Jiang did teach such an element, there would have been no motivation to combine Traversat and Jiang, since the intended purpose of Traversat to avoid use of local trees of an application would be destroyed. Thus, appellants respectfully submit that the combination is improper.

Further, there is no suggestion in the references themselves to combine the references, as proposed. Thus, again, appellants respectfully submit that the combination is improper.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

Based on the foregoing, appellants respectfully submit that the combination of Traversat and Jiang fails to describe, teach or suggest at least one element of the claimed invention, as recited in claim 1, as well as the other independent claims. Moreover, appellants respectfully submit that the combination of Traversat and Jiang is improper.

For all the reasons stated above, appellants respectfully submit that their invention is patentable over the combination of Traversat and Jiang. Therefore, appellants respectfully submit that the rejection of the claims of Group I, Subgroup (i) under 35 U.S.C. 103(a) is erroneous, and appellants respectfully request reversal of the rejection. In addition, appellants respectfully request reversal of the rejection of the claims of Subgroup (ii), because each of the claims of Subgroup (ii) depends from one of the independent claims of Subgroup (i) of Group I.

Group I, Subgroup (ii): Claims 12, 18 & 24

Appellants respectfully submit that claims 12, 18, and 24 have a separate basis of patentability from the claims of Subgroup (i), and that the rejection of the claims of this subgroup is erroneous for the reasons discussed below.

For example, claim 12 further recites that the locked resource is used by another thread of the multi-threaded client application. In other words, even though a lock on a resource has been obtained for one thread of a multi-threaded client application, another thread of the multi-threaded client application can use that same resource concurrently, as disclosed in appellants' specification on page 15, lines 4-20 and in FIG. 6, for example. This other thread accesses the resource via a mount point of the local tree of the client application.

The final Office Action cites col. 8, lines 21-24 and lines 63-67 of Traversat as teaching this aspect of appellants' invention. Appellants respectfully disagree with this characterization of Traversat. A closer reading of the cited portion of Traversat reveals that Traversat describes

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

notifying any waiting threads when locks on nodes in a database are released at the conclusion of an attempted transaction which updates the database. The notification of waiting threads that locks on a database have been released in Traversat implies that the threads access the database sequentially. This is contrary to appellants' invention as recited in claim 12, wherein a lock on a resource is obtained by a thread of a multi-threaded client application; and another thread of the multi-threaded client application concurrently uses the same resource. Thus, appellants respectfully submit that Traversat fails to describe, teach or suggest this element of their claimed invention.

Since the final Office Action does not indicate that Jiang cures this deficiency, appellants respectfully submit that a prima facie case of obviousness has not been provided for claim 12.

The arguments presented above for claim 12 apply to the analogous system and program storage device claims 18 and 24, respectively. Accordingly, appellants respectfully request reversal of the rejection of the claims of Subgroup (ii) of Group I.

Conclusion

Appellants respectfully submit that their claimed invention would not have been obvious to one of ordinary skill in the art based upon Traversat and Jiang, either alone or in combination.

By way of example, appellants respectfully submit that the combination of Traversat and Jiang, either alone or in combination, does not describe, teach or suggest employing a local tree, which is local to a multithreaded client application, in obtaining a lock of a resource of a global data repository. That is, each reference fails to teach or suggest the use of a tree local to a client application in obtaining a lock. Since both references fail to teach or suggest this claimed element, the combination also fails to teach or suggest this claimed element.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

For all of the reasons hereinabove, appellants respectfully submit that the rejection of their claims under 35 U.S.C. 103 is erroneous. Accordingly, reversal of the §103 rejection is respectfully requested.

Blanche E. Schiller
Blanche E. Schiller
Attorney for Appellants
Registration No.: 35,670

Dated: July 30, 2004.

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

Appendix A

1. A method of managing locking of resources of a global data repository of a distributed computing environment, said method comprising:

issuing a request, via a thread of a multithreaded client application of said distributed computing environment, for a lock of a resource of said global data repository; and

obtaining said lock for said thread independent of a threading model of an operating system of said distributed computing environment,

wherein said obtaining comprises employing a local tree in obtaining said lock, said local tree being local to the client application and having a mount point usable by the client application to lock said resource, and wherein said resource is further lockable via another mount point of one of said local tree and another local tree.

3. A system of managing locking of resources of a global data repository of a distributed computing environment, said system comprising:

means for issuing a request, via a thread of a multithreaded client application of said distributed computing environment, for a lock of a resource of said distributed global data repository; and

means for obtaining said lock for said thread independent of a threading model of an operating system of said distributed computing environment,

wherein said means for obtaining comprises means for employing a local tree in obtaining said lock, said local tree being local to the client application and having a mount point usable by the client application to lock said resource, and wherein said

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

resource is further lockable via another mount point of one of said local tree and another local tree.

5. At least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of managing locking of resources of a global data repository of a distributed computing environment, said method comprising:

issuing a request, via a thread of a multithreaded client application of said distributed computing environment, for a lock of a resource of said global data repository; and

obtaining said lock for said thread independent of a threading model of an operating system of said distributed computing environment,

wherein said obtaining comprises employing a local tree in obtaining said lock, said local tree being local to the client application and having a mount point usable by the client application to lock said resource, and wherein said resource is further lockable via another mount point of one of said local tree and another local tree.

7. The method of claim 1, wherein the employing comprises connecting the local tree to a server data tree.

8. The method of claim 7, wherein the connecting comprises connecting the local tree to the server data tree via a mount point on the local tree.

9. The method of claim 1, wherein the issuing comprises issuing a request for a lock of at least one table of the global data repository.

10. The method of claim 1, wherein the issuing comprises issuing the request from a server associated with said resource.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

11. The method of claim 1, further comprising unlocking said resource by the thread of the multithreaded client application.

12. The method of claim 1, further comprising using said resource by another thread of the multithreaded client application.

13. The system of claim 3, wherein the means for employing comprises means for connecting the local tree to a server data tree.

14. The system of claim 13, wherein the means for connecting comprises means for connecting the local tree to the server data tree via a mount point on the local tree.

15. The system of claim 3, wherein the means for issuing comprises means for issuing a request for a lock of at least one table of the global data repository.

16. The system of claim 3, wherein the means for issuing comprises means for issuing the request from a server associated with said resource.

17. The system of claim 3, further comprising means for unlocking said resource by the thread of the multithreaded client application.

18. The system of claim 3, further comprising means for using said resource by another thread of the multithreaded client application.

19. The at least one program storage device of claim 5, wherein the employing comprises connecting the local tree to a server data tree.

20. The at least one program storage device of claim 19, wherein the connecting comprises connecting the local tree to the server data tree via a mount point on the local tree.

21. The at least one program storage device of claim 5, wherein the issuing comprises issuing a request for a lock of at least one table of the global data repository.

Appellants: Uceda-Sosa et al.
Serial No.: 09/584,609
Filing Date: 05/31/2000

22. The at least one program storage device of claim 5, wherein the issuing comprises issuing the request from a server associated with said resource.

23. The at least one program storage device of claim 5, further comprising unlocking said resource by the thread of the multithreaded client application.

24. The at least one program storage device of claim 5, further comprising using said resource by another thread of the multithreaded client application.

* * * * *